Search
Go

Resource Centers
Bioinformatics
C/C++
Databases
Digital Media
Enterprise Development
Game Development
Java
Linux/Unix
Macintosh/OS X
.NET
Open Source
Oracle
Perl
Python
Scripting
Security
SysAdmin/Networking
Web
Web Services
Windows
Wireless
XML

Book Series
CD Bookshelves
Cookbooks
Developer's Notebooks
Hacks
Head First
In A Nutshell
Missing Manuals
Pocket References
Technology & Society

Publishing Partners
No Starch Press
Paraglyph Press
Pragmatic Bookshelf
SitePoint
Syngress Publishing

Online Publications
LinuxDevCenter.com
MacDevCenter.com
ONDotnet.com
ONJava.com
ONLamp.com
OpenP2P.com
Perl.com

Search Results

1 - 10 of 36 total results for query **transparent overlay** sorted by **score** in **"xml"**.

Sort by Date | Hide Summaries | Next 10

New Search Advanced Search

XML.com: Creative Comments: On the Uses and Abuses of Markup
By Kendall Grant Clark
The way Creative Commons recommends linking its machine-readable licenses into HTML little sense, says Kendall Clark, and proposes alternatives.
Context: ...a fantastic and impractical threat, it will not be a separate web but, rather, **overla** existing one. The Semantic Web isn't a replacement, it's a...
[January 15, 2003]

XML.com: News From the Expo Floor
By Simon St. Laurent
Reviewing the Expo Floor from XML DevCon 2000, we take a look at a couple of new kids Numerator and XMLMATE.
Context: ...and charting functions, allowing data set designers and users to create **overlays** o information, combining information from different sources and...
[June 27, 2000]

XML.com: Handling Binary Data in XML Documents
By Lisa Rein
Binary data can present some interesting problems. This article looks at ways to support bina as images in XML documents.
Context: ...in the same stream as the vector data because it might be an **overlay** on the pixel **overlay** may not have any logical meaning by itself. As...
[July 24, 1998]

XML.com: Getting Started with XML Programming
By Norman Walsh
How is processing an XML document really different than processing a plain old text file?
Context: ...our own versions of getProfileString() and setProfileString() that provide **transp** to XML configuration files. A simple configuration file...
[April 21, 1999]

XML.com: ISO to Require Royalties?
By Kendall Grant Clark
The ISO, a worldwide standards body, is proposing to charge fees for commercial usage in s their standardized country, language and currency codes. This would have a wide-ranging ne on the infrastructure of the web and related ...
Context: ...the W3C's notes and recommendations (not to mention both organizations fairly

WebServices.XML.com
WindowsDevCenter.com
XML.com

Special Interest
Ask Tim
Beta Chapters
Events
From the Editors List
Letters
Open Books
tim.oreilly.com

Special Sales
Academic
Corporate Services
Government

Inside O'Reilly
About O'Reilly
Bookstores
Catalog Request
Contact Us
International
User Groups
Writing for O'Reilly

work group product policies), the ISO's standards documents...
[September 24, 2003]

XML.com: Using libferris with XML
By Ben Martin
The libferris library is a hierarchical data interface, providing uniform access to relational da
the filesystem. This article explores the possibilities of its use with XML.
Context: ...cannot be stored in the filesystem itself it's stored locally as RDF and made **tran**
accessible again for the file that it's attached to. Such a...
[March 31, 2004]

XML.com: SVG: Where Are We Now?
By Antoine Quint
SVG expert Antoine Quint surveys the current state of tool support for the W3C's Scalable V
Graphics Recommendation.
Context: ...Viewer SVG support allow for some quite advanced compositions (e.g., semi-**tr**
SVG shapes over a video clip). This new version also comes with a...
[November 21, 2001]

XML.com: Browser Lockouts and Monopoly Power
By Kendall Grant Clark
Last week's controversial blocking of certain browsers by MSN.com was excused by means
appeal to standards compliance. Kendall Clark reporst on the debate and the possible implica
Microsoft antitrust negotiations.
Context: ...enough, Dumbill pointed out, Microsoft was trying to justify it by offering **trans**
false claims about public standards compliance. As Dumbill...
[October 31, 2001]

XML.com: Nobody REALLY Asked Me, But...
By John E. Simpson
On the second anniversary of his column, John Simpson returns to the question of obscuring
of an XML document, exploring a good deal of XSLT along the way.
Context: ...the numbers -- including the all-important credit card account -- are plainly **tran**
Before tackling James's question outright, consider some of...
[August 28, 2002]

XML.com: Introducing PyXML
By Uche Ogbuji
In the second Python and XML column, Uche Ogbuji introduces PyXML, the add-on XML l
builds upon Python's core XML support.
Context: ...PyXML **overlays** the xml module that comes with Python. It replaces all the con
Python module with its own versions. This is usually OK...
[September 25, 2002]

O'Reilly Home | Privacy Policy

Subscribe (Full Service) Register (Limited Service, Free) Login

PORTAL

Search: ⦿ The ACM Digital Library ○ The Guide

US Patent & Trademark Office

transparent overlay SEARCH

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used transparent overlay

Found **4,809** of **150,138**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

☐ Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200

Best 200 shown

Result page: **1** 2 3 4 5 6 7 8 9 10 next

Relevance scale

1 Using transparent props for interaction with the virtual table

Dieter Schmalstieg, L. Miguel Encarnação, Zsolt Szalavári

April 1999 **Proceedings of the 1999 symposium on Interactive 3D graphics**

Full text available: pdf(659.17 KB) Additional Information: full citation, references, citings, index terms

2 A proposal to establish a pseudo virtual memory via writable overlays

Thomas R. Spacek

June 1972 **Communications of the ACM**, Volume 15 Issue 6

Full text available: pdf(595.40 KB) Additional Information: full citation, abstract, references

Many computer systems solve executable storage size problems for large programs by using overlays. However, it appears that no one overlay scheme contains a well-balanced combination of the most useful capabilities which are found in various existing techniques. A proposal is presented which utilizes several of the best capabilities from existing schemes and is complemented by several additional features, e.g. writable overlays. The writable overlay capability provides a virtual memory effe ...

Keywords: collector, folding, linkage editor, loader, overlay, overlay structure, paging, segment, segmentation, virtual memory

3 An experimental evaluation of transparent user interface tools and information content

Beverly L. Harrison, Gordon Kurtenbach, Kim J. Vicente

December 1995 **Proceedings of the 8th annual ACM symposium on User interface and software technology**

Full text available: pdf(1.28 MB) Additional Information: full citation, references, citings, index terms

Keywords: display design, interaction technology, toolglass, transparency, user interface design

4 Transparent layered user interfaces: an evaluation of a display design to enhance focused and divided attention

Beverly L. Harrison, Hiroshi Ishii, Kim J. Vicente, William A. S. Buxton

May 1995 **Proceedings of the SIGCHI conference on Human factors in computing systems**
Full text available: html(44.09 KB) Additional Information: full citation, references, citings, index terms

5 The application of scene synthesis techniques to the display of multidimensional image data
Philip K. Robertson, John F. O'Callaghan
October 1985 **ACM Transactions on Graphics (TOG)**, Volume 4 Issue 4
Full text available: pdf(4.42 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Superimposition of two image data sets allows the spatial distribution of one to be directly related to that of the other. If the two data sets have different spatial structures, the composite image is generally confusing and difficult to interpret. A method of representing image data sets in the form of naturally occurring variables in a realistic apparently three-dimensional scene is presented. One data set is represented by the topography of a surface, depicted by shaded-relief methods, ...

6 TeamWorkStation: towards a seamless shared workspace
H. Ishii
September 1990 **Proceedings of the 1990 ACM conference on Computer-supported cooperative work**
Full text available: pdf(1.48 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper introduces TeamWorkStation (TWS), a new desktop real-time shared workspace characterized by reduced cognitive seams. TWS integrates two existing kinds of individual workspaces, computers and desktops, to create a virtual shared workspace. The key ideas are the overlay of individual workspace images in a virtual shared workspace and the creation of a shared drawing surface. Because each co-worker can continue to use his/her favorite application programs or manual tools in the virt ...

7 GROUPKIT: a groupware toolkit for building real-time conferencing applications
Mark Roseman, Saul Greenberg
December 1992 **Proceedings of the 1992 ACM conference on Computer-supported cooperative work**
Full text available: pdf(938.94 KB) Additional Information: full citation, references, citings, index terms

Keywords: development tools, real-time groupware, toolkit

8 An experimental evaluation of transparent menu usage
Beverly L. Harrison, Kim J. Vicente
April 1996 **Proceedings of the SIGCHI conference on Human factors in computing systems: common ground**
Full text available: pdf(1.54 MB) html(36.63 KB) Additional Information: full citation, references, citings, index terms

Keywords: display design, evaluation, interaction technology, toolglass, transparency, user interface design

9 A window-based graphics frame store architecture
Richard J. Westmore
October 1988 **ACM Transactions on Graphics (TOG)**, Volume 7 Issue 4
Full text available: pdf(1.02 MB) Additional Information: full citation, abstract, references, index terms, review

A proposal for a scalable frame store architecture for a hardware-based window graphics system is described. The architecture is based on a distributed linear array of common elements called microframe stores. Each window and viewport can be independently configured in size and color depth. Unlike the strip-based hardware window systems referenced, this system places no restrictions on the number of transitions that can be accommodated on each scanline. The approach described allows real-ti ...

10 Integration of interpersonal space and shared workspace: ClearBoard design and experiments
Hiroshi Ishii, Minoru Kobayashi, Jonathan Grudin
October 1993 **ACM Transactions on Information Systems (TOIS)**, Volume 11 Issue 4
Full text available: pdf(2.91 MB) Additional Information: full citation, abstract, references, citings, index terms

We describe the evolution of the novel shared drawing medium ClearBoard which was designed to seamlessly integrate an interpersonal space and a shared workspace. ClearBoard permits coworkers in two locations to draw with color markers or with electronic pens and software tools while maintaining direct eye contact and the ability to employ natural gestures. The ClearBoard design is based on the key metaphor of "talking through and drawing on a transparent glass window." We descri ...

Keywords: eye contact, gaze awareness, gaze direction, groupware, seamless design, shared drawing, video conference

11 MMVIS: design and implementation of a multimedia visual information seeking environment
Stacie Hibino, Elke A. Rundensteiner
February 1997 **Proceedings of the fourth ACM international conference on Multimedia**
Full text available: pdf(1.70 MB) Additional Information: full citation, references, citings, index terms

Keywords: interactive visualizations, multimedia system design, temporal visual query language, video analysis

12 Tools and technology I: Awareness support in a groupware widget toolkit
Jason Hill, Carl Gutwin
November 2003 **Proceedings of the 2003 international ACM SIGGROUP conference on Supporting group work**
Full text available: pdf(490.91 KB) Additional Information: full citation, abstract, references, index terms

Group awareness is an important part of synchronous collaboration, and support for group awareness can greatly improve groupware usability. However, it is still difficult to build groupware that supports group awareness. To address this problem, we have developed the MAUI toolkit, a Java toolkit with a broad suite of awareness-enhanced UI components. The toolkit contains both extensions of standard Swing widgets, and groupware-specific components such as telepointers. All components have added f ...

Keywords: awareness, feedthrough, group widgets, groupware interfaces

13 Vertical handoffs in wireless overlay networks
Mark Stemm, Randy H. Katz
December 1998 **Mobile Networks and Applications**, Volume 3 Issue 4
Full text available: pdf(770.58 KB) Additional Information: full citation, abstract, references, citings, index terms

No single wireless network technology simultaneously provides a low latency, high bandwidth, wide area data service to a large number of mobile users. Wireless Overlay Networks - a hierarchical structure of room-size, building-size, and wide area data networks - solve the problem of providing network connectivity to a large number of mobile users in an efficient and scalable way. The specific topology of cells and the wide variety of network technologies that comprise wireless o ...

14 Peer to peer networks: Tarzan: a peer-to-peer anonymizing network layer
Michael J. Freedman, Robert Morris
November 2002 **Proceedings of the 9th ACM conference on Computer and communications security**
Full text available: pdf(242.72 KB) Additional Information: full citation, abstract, references, citings, index terms

Tarzan is a peer-to-peer anonymous IP network overlay. Because it provides IP service, Tarzan is general-purpose and transparent to applications. Organized as a decentralized peer-to-peer overlay, Tarzan is fault-tolerant, highly scalable, and easy to manage.Tarzan achieves its anonymity with layered encryption and multi-hop routing, much like a Chaumian mix. A message initiator chooses a path of peers pseudo-randomly through a restricted topology in a way that adversaries cannot easily influenc ...

Keywords: IP tunnels, anonymity, cover traffic, distributed trust, mix-nets, overlay networks, peer-to-peer

15 Advancing interaction: DeepDocument: use of a multi-layered display to provide context awareness in text editing
Masood Masoodian, Sam McKoy, Bill Rogers, David Ware
May 2004 **Proceedings of the working conference on Advanced visual interfaces**
Full text available: pdf(329.78 KB) Additional Information: full citation, abstract, references, index terms

Word Processing software usually only displays paragraphs of text immediately adjacent to the cursor position. Generally this is appropriate, for example when composing a single paragraph. However, when reviewing or working on the layout of a document it is necessary to establish awareness of current text in the context of the document as a whole. This can be done by scrolling or zooming, but when doing so, focus is easily lost and hard to regain.We have developed a system called DeepDocument us ...

Keywords: Deep Video™, Microsoft Word™, context awareness, multi-layered display, text editing, word processing

16 Context and interaction in zoomable user interfaces
Stuart Pook, Eric Lecolinet, Guy Vaysseix, Emmanuel Barillot
May 2000 **Proceedings of the working conference on Advanced visual interfaces**
Additional Information: full citation, abstract, references, citings, index

Full text available: pdf(3.68 MB) terms

Zoomable User Interfaces (ZUIs) are difficult to use on large information spaces in part because they provide insufficient context. Even after a short period of navigation users no longer know where they are in the information space nor where to find the information they are looking for. We propose a temporary in-place context aid that helps users position themselves in ZUIs. This context layer is a transparent view of the context that is drawn over the users' focus of atte ...

17 Handling visual media preparation for Ohio state's computer center workshops

Gail Peters
November 1984 **Proceedings of the 12th annual ACM SIGUCCS conference on User services**

Full text available: pdf(398.02 KB) Additional Information: full citation, abstract, references, index terms

At The Ohio State University, approximately 20 general-purpose workshops are provided each quarter to service the needs of our user community. We have found it useful to organize the teaching materials for these workshops by standardizing the overhead transparencies that are used by our instructors. We have found that as a direct benefit of this team approach, we can maintain consistent organization in teaching our workshops; our instructors are certain to cover the important aspects of eac ...

18 Three-dimensional medical imaging: algorithms and computer systems

M. R. Stytz, G. Frieder, O. Frieder
December 1991 **ACM Computing Surveys (CSUR)**, Volume 23 Issue 4

Full text available: pdf(7.38 MB) Additional Information: full citation, references, citings, index terms, review

Keywords: Computer graphics, medical imaging, surface rendering, three-dimensional imaging, volume rendering

19 An experimental system for creating and presenting interactive graphical documents

S. Feiner, S. Nagy, A. van Dam
January 1982 **ACM Transactions on Graphics (TOG)**, Volume 1 Issue 1

Full text available: pdf(3.53 MB) Additional Information: full citation, references, citings, index terms

Keywords: maintenance and repair, pictorial information systems

20 Technical opinion: getting the best of both real and virtual worlds

Eric W. Tatham
September 1999 **Communications of the ACM**, Volume 42 Issue 9

Full text available: pdf(108.88 KB) html(14.52 KB) Additional Information: full citation, references, index terms

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

Your search matched **29** of **1123491** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enterin new one in the text box.

transparent<and>overlay Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

1 **Value of fiber overlays in WDM metro networks**
Farjady, F.; Antoniades, N.; Wagner, R.E.; Yadlowsky, M.J.;
Photonics Technology Letters, IEEE , Volume: 15 , Issue: 2 , Feb. 2003
Pages:329 - 331

[Abstract] [PDF Full-Text (255 KB)] IEEE JNL

2 **Transparent query caching in peer-to-peer overlay networks**
Patro, S.; Hu, Y.C.;
Parallel and Distributed Processing Symposium, 2003. Proceedings. International , 22-26 April 2003
Pages:10 pp.

[Abstract] [PDF Full-Text (315 KB)] IEEE CNF

3 **Computing layered surface representations: an algorithm for detecti and separating transparent overlays**
Singh, M.; Xiaolei Huang;
Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Comp Society Conference on , Volume: 2 , 18-20 June 2003
Pages:II - 11-18 vol.2

[Abstract] [PDF Full-Text (1095 KB)] IEEE CNF

4 **Scalable diverse protected multicast as a programmable service leveraging IP-multicast**
Bachmeir, C.; Tabery, P.;
Network Computing and Applications, 2003. NCA 2003. Second IEEE Internati Symposium on , 16-18 April 2003
Pages:97 - 104

[Abstract] [PDF Full-Text (470 KB)] IEEE CNF

---

5 **Overlays and churn in WDM interconnected-ring metro networks**
Farjady, F.; Antoniades, N.A.; Wagner, R.E.; Yadlowsky, M.J.;
Lasers and Electro-Optics Society, 2001. LEOS 2001. The 14th Annual Meeting the IEEE , Volume: 2 , 12-13 Nov. 2001
Pages:431 - 432 vol.2

[Abstract] [PDF Full-Text (148 KB)] IEEE CNF

---

6 **Toward wide-scale all-optical transparent networking: the ACTS opt pan-European network (OPEN) project**
Chbat, M.W.; Grard, E.; Berthelon, L.; Jourdan, A.; Perrier, P.A.; Leclert, A.; Landousies, B.; Ramdane, A.; Parnis, N.; Jones, E.V.; Limal, E.; Poulsen, H.N. Pedersen, R.J.S.; Flaaronning, N.; Vercauteren, D.; Puleo, M.; Ciaramella, E.; Marone, G.; Hess, R.; Melchior, H.; Parys, W.V.; Demeester, P.M.; Godsvang, Olsen, T.; Hjelme, D.R.;
Selected Areas in Communications, IEEE Journal on , Volume: 16 , Issue: 7 , S 1998
Pages:1226 - 1244

[Abstract] [PDF Full-Text (476 KB)] IEEE JNL

---

7 **Investigation of optical fibre switch using electro-optic interlays**
McCallion, K.; Johnstone, W.; Thursby, G.;
Electronics Letters , Volume: 28 , Issue: 4 , 13 Feb. 1992
Pages:410 - 411

[Abstract] [PDF Full-Text (160 KB)] IEE JNL

---

8 **Optical cost metrics in multi-layer traffic engineering for IP-over-op networks**
Puype, B.; Qiang Yan; De Maesschalck, S.; Colle, D.; Pickavet, M.; Demeester
Transparent Optical Networks, 2004. Proceedings of 2004 6th International Conference on , Volume: 1 , 4-8 July 2004
Pages:75 - 80 vol.1

[Abstract] [PDF Full-Text (460 KB)] IEEE CNF

---

9 **Proxy location problems and their generalizations**
Sumi Choi; Shavitt, Y.;
Distributed Computing Systems Workshops, 2003. Proceedings. 23rd Internat Conference on , 19-22 May 2003
Pages:898 - 904

[Abstract] [PDF Full-Text (372 KB)] IEEE CNF

---

10 **Breaking the copy/paste cycle: the Stretchable Selection Tool**
Apperley, M.; Fletcher, D.; Rogers, B.;
User Interface Conference, 2000. AUIC 2000. First Australasian , 31 Jan.-3 Fe 2000
Pages:3 - 10

[Abstract] [PDF Full-Text (228 KB)] **IEEE CNF**

---

11 **The optical Pan-European network (ACTS project OPEN)**
Chbat, M.W.;
Advanced Applications of Lasers in Materials Processing, 1996/Broadband Opti Networks/Smart Pixels/Optical MEMs and Their Applications, 1996. IEEE/LEOS 1996 Summer Topical Meetings: , 5-9 Aug. 1996
Pages:63 - 64

[Abstract] [PDF Full-Text (128 KB)] **IEEE CNF**

---

12 **Traffic management and network planning for optical IP over WDM networks**
Gagnaire, M.;
Transparent Optical Networks, 2001. Proceedings of 2001 3rd International Conference on , 18-21 June 2001
Pages:163 - 172

[Abstract] [PDF Full-Text (780 KB)] **IEEE CNF**

---

13 **Information compression by MR-tissue type imaging**
Bielke, G.; Jungke, M.; Meindl, S.;
Engineering in Medicine and Biology Society, 1989. Images of the Twenty-Firs Century. Proceedings of the Annual International Conference of the IEEE Engineering in , 9-12 Nov. 1989
Pages:522 - 523 vol.2

[Abstract] [PDF Full-Text (144 KB)] **IEEE CNF**

---

14 **Closed loop response of nonlinear systems by a functional transformation approach**
Hill, J.;
Automatic Control, IRE Transactions on , Volume: 7 , Issue: 4 , Jul 1962
Pages:39 - 44

[Abstract] [PDF Full-Text (472 KB)] **IEEE JNL**

---

15 **The Locus of Points of Constant VSWR When Renormalized to a Different Characteristic Impedance (Letters)**
Malherbe, J.A.G.;
Microwave Theory and Techniques, IEEE Transactions on , Volume: 25 , Issue: 5 , May 1977
Pages:444 - 445

[Abstract] [PDF Full-Text (192 KB)] **IEEE JNL**

---

1 2 **Next**

Web Results **1** - **10** of about **20,100** for **transparent overlay HTML XML**. (**0.49** seconds)

XML.com: An SVG Case Study: Integrated, Dynamic Avalanche ...
... box, adding an image and a **transparent overlay** to create ... www.webstandards.org/about/ [3] http://**xml**.apache.org ... www.w3.org/TR/SVG11/text.**html**#TextAnchorProperty: ...
www.**xml**.com/pub/a/2003/04/23/avalanche.**html**?page=last - 43k - Cached - Similar pages

GIS map server integration
... User definable **see-through** map shading is ideal for ... request, and remove any **transparent** colors set ... the server follows NCS_OVERLAY_FAILURE alert("**Overlay** Error\r ...
www.geoimage.com.au/IWSDoc/ **html**/GIS_Map_server_integration.htm - 62k - Cached - Similar pages

Image Maker **XML** Reference
... "gif" or "jpg". Default is "gif" **transparency**—(optional) if "true" gif **transparency** is used. bgcolor is set to **transparent**. ... <imgop type="**overlay**"> ... ...
www.psoft.net/ssdoc/ImageMaker-XML-reference.**html** - 30k - Cached - Similar pages

jmac.org
... map used in this way to a **transparent overlay** tossed over ... without yours having to paw through every **HTML** file to ... and after two full days of **XML** head-crammage ...
www.jmac.org/writings/**xml**2000.**html** - 33k - Cached - Similar pages

[PDF] Net-Top-Box
File Format: PDF/Adobe Acrobat - View as HTML
... Still Background **Transparent Overlay** MPEG Video Flash Animation Scrolling ... STB Internet For **HTML** Media & News Text ... supplier can easily edit the **XML** script to ...
www.hollycroftassociates.com/ images/ONELAN_NTB_DATA_SHEET_APR2004.pdf - Similar pages

[PDF] Adobe InDesign 2.0 Overview
File Format: PDF/Adobe Acrobat - View as HTML
... they need to reliably output **transparency** effects. ... layers, which work like one or more **transparent overlays**. ... 4 http://www.adobe.com/products/indesign/main.**html** ...
www.pacific.adobe.com/education/pdf/id_overview.pdf - Similar pages

SVG <image> tag fails: 'xlink:href' required
... need to > **overlay** a number of **transparent** > images ... 740" height="499"/> > <image xlink:href="**overlay**.png" > x ... apache.org/cocoon/faq/index.**html**> > > To unsubscribe ...
archives.real-time.com/pipermail/ cocoon-users/2002-October/023692.**html** - 7k - Cached - Similar pages

WebCGM vector graphics
... This is known as the **overlay** model of object ... a continuum between fully opaque and fully **transparent**) can be defined via an alpha-**transparency** Escape element. ...
xml.coverpages.org/oasis-webcgm1.**html** - 45k - Cached - Similar pages

Popup **overlay** technique from CBSNews (BarelyBlogging)
... Popup **overlay** technique from CBSNews. ... image as a background which shows through the **transparent** parts ... automatic, e-mail address never displayed, **HTML** allowed: <a ...
www.barelyfitz.com/blog/archives/2004/11/03/114/ - 19k - Cached - Similar pages

bobbyvandersluis.com | the best of the Web

... Cross-browser semi-**transparent** backgrounds; Making Internet ... display of PNG Alpha **Transparency** with MSIE (PhP ... Working Group (WHAT WG); **HTML Overlays**; Weblications; ...
www.bobbyvandersluis.com/main/linkLibrary.php - 59k - Feb 1, 2005 - Cached - Similar pages

Gooooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

transparent overlay HTML XML [Search]

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

Web

Results **1** - **10** of about **240,000** for **transparent overlay**. (**0.58** seconds)

Transparent overlay in Flash
Transparent Overlay with Flash (advanced). See what I'm talking about (IE6+): **transparent overlay** example. ... More examples of Flash/**transparent overlays**: ...
www.webworksite.com/**overlay**.shtml - 36k - Cached - Similar pages

Solaris X Window System Developer's Guide
... **overlay** extension allows the creation and manipulation of **transparent overlay** windows. These windows are X windows that allow the user to **see through** to the ...
docsun.cites.uiuc.edu/sun_docs/ C/solaris_9/SUNWdev/OWPG/p14.html - 19k - Cached - Similar pages

Solaris X Window System Developer's Guide
... Server **Overlays** Versus **Transparent Overlays**. ... The **Transparent Overlay** Extension is a Sun proprietary method to provide **overlay** capability in the X Window System. ...
docsun.cites.uiuc.edu/sun_docs/ C/solaris_9/SUNWdev/OWPG/p13.html - 13k - Cached - Similar pages
[More results from docsun.cites.uiuc.edu]

docs.sun.com: Solaris X Window System Developer's Guide
... Solaris 10 Software Developer Collection >> Solaris X Window System Developer's Guide >> 6. **Transparent Overlay** Windows. ... Chapter 6 **Transparent Overlay** Windows. ...
docs.sun.com/app/docs/doc/816-0279/6m6pd1d00?a=view - 13k - Cached - Similar pages

docs.sun.com: Solaris X Window System Developer's Guide
... Solaris 2.5 Software Developer AnswerBook >> Solaris X Window System Developer's Guide >> 5 **Transparent Overlay** Windows. ... **Transparent Overlay** Windows. 5. ...
docs.sun.com/app/docs/doc/802-1951/6i5ut5rfb?a=view - 13k - Cached - Similar pages
[More results from docs.sun.com]

Smaller Animals Discussion Forums - Viewing topic #13 - can't get ...
Subject: "can't get **transparent overlay** to work", ... sg_giftpuzzle, Tue Dec-10-02 06:49 AM. #13, "can't get **transparent overlay** to work", ...
www.smalleranimals.com/dc/dcboard.php?az=show_ topic&forum=111&topic_id=13&mesg_id=13&page= - 16k - Cached - Similar pages

Smaller Animals Discussion Forums - "can't get **transparent overlay** ...
Smaller Animals Discussion Forums. "can't get **transparent overlay** to work". ... AM (EST). "can't get **transparent overlay** to work". I'm trying ...
www.smalleranimals.com/cgi-bin/dcforum/ dcboard.cgi? az=read_count&om=12&forum=DCForumID7 - 22k - Cached - Similar pages
[More results from www.smalleranimals.com]

Overlay Plane **Transparency** and the X Windows System
... An **overlay** visual's **transparency** feature enables you to render opaque objects (for example, menus and text) to a **transparent overlay** window and at the same ...
docs.hp.com/en/B2355-90143/ch04s10.html - 37k - Cached - Similar pages

Large Background
... height is particularly tricky, if you make it too small, text will flow "underneath" the **transparent overlay**, too largen and there's a large amount of blank ...

www.scit.wlv.ac.uk/encyc/bdemo3.html - 4k - Feb 1, 2005 - Cached - Similar pages

Baby - **Transparent Overlay** from Memories Complete
... Templates. Shop By Brand Shop By Theme Sale Items Clearance Items. Baby - **Transparent Overlay** from Memories Complete. $2.98 MC123 Quantity. ...
www.cropaholics.com/product7922.html - 13k - Cached - Similar pages

Gooooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

Free! Google Desktop Search: Search your own computer.

transparent overlay Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

Google

Web Images Groups News Froogle Local New! more »

transparent overlay HTML XML [Search] Advanced Search Preferences

Web Results **1** - **10** of about **20,200** for **transparent overlay HTML XML**. (**0.20** seconds)

XML.com: An SVG Case Study: Integrated, Dynamic Avalanche ...
... box, adding an image and a **transparent overlay** to create ... www.webstandards.org/about/ [3] http://**xml**.apache.org ... www.w3.org/TR/SVG11/text.**html**#TextAnchorProperty: ...
www.**xml**.com/pub/a/2003/04/23/avalanche.**html**?page=last - 43k - Cached - Similar pages

Image Maker **XML** Reference
... "gif" or "jpg". Default is "gif" **transparency**—(optional) if "true" gif **transparency** is used. bgcolor is set to **transparent**. ... <imgop type="**overlay**"> ... ...
www.psoft.net/ssdoc/ImageMaker-XML-reference.**html** - 30k - Cached - Similar pages

jmac.org
... map used in this way to a **transparent overlay** tossed over ... without yours having to paw through every **HTML** file to ... and after two full days of **XML** head-crammage ...
www.jmac.org/writings/**xml**2000.**html** - 33k - Cached - Similar pages

[PDF] Net-Top-Box
File Format: PDF/Adobe Acrobat - View as HTML
... Still Background **Transparent Overlay** MPEG Video Flash Animation Scrolling ... STB Internet For **HTML** Media & News Text ... supplier can easily edit the **XML** script to ...
www.hollycroftassociates.com/ images/ONELAN_NTB_DATA_SHEET_APR2004.pdf - Similar pages

GIS map server integration
... User definable **see-through** map shading is ideal for ... request, and remove any **transparent** colors set ... the server follows NCS_OVERLAY_FAILURE alert("**Overlay** Error\r ...
www.geoimage.com.au/IWSDoc/ **html**/GIS_Map_server_integration.htm - 62k - Cached - Similar pages

[PDF] Adobe InDesign 2.0 Overview
File Format: PDF/Adobe Acrobat - View as HTML
... they need to reliably output **transparency** effects. ... layers, which work like one or more **transparent overlays**. ... 4 http://www.adobe.com/products/indesign/main.**html** ...
www.pacific.adobe.com/education/pdf/id_overview.pdf - Similar pages

SVG <image> tag fails: 'xlink:href' required
... need to > **overlay** a number of **transparent** > images ... 740" height="499"/> > <image xlink:href="**overlay**.png" > x ... apache.org/cocoon/faq/index.**html**> > > To unsubscribe ...
archives.real-time.com/pipermail/ cocoon-users/2002-October/023692.**html** - 7k - Cached - Similar pages

WebCGM vector graphics
... This is known as the **overlay** model of object ... a continuum between fully opaque and fully **transparent**) can be defined via an alpha-**transparency** Escape element. ...
xml.coverpages.org/oasis-webcgm1.**html** - 45k - Cached - Similar pages

Popup **overlay** technique from CBSNews (BarelyBlogging)
... Popup **overlay** technique from CBSNews. ... image as a background which shows through the **transparent** parts ... automatic, e-mail address never displayed, **HTML** allowed: <a ...
www.barelyfitz.com/blog/archives/2004/11/03/114/ - 19k - Cached - Similar pages

bobbyvandersluis.com | the best of the Web

... Cross-browser semi-**transparent** backgrounds; Making Internet ... display of PNG Alpha **Transparency** with MSIE (PhP ... Working Group (WHAT WG); **HTML Overlays**; Weblications; ...
www.bobbyvandersluis.com/main/linkLibrary.php - 59k - Feb 1, 2005 - Cached - Similar pages

transparent overlay HTML XML | Search

An Improved Error Model for Noisy Channel Spelling Correction

Eric Brill and Robert C. Moore
Microsoft Research
One Microsoft Way
Redmond, Wa. 98052
{brill,bobmoore}@microsoft.com

Abstract

The noisy channel model has been applied to a wide range of problems, including spelling correction. These models consist of two components: a source model and a channel model. Very little research has gone into improving the channel model for spelling correction. This paper describes a new channel model for spelling correction, based on generic string to string edits. Using this model gives significant performance improvements compared to previously proposed models.

Introduction

The noisy channel model (Shannon 1948) has been successfully applied to a wide range of problems, including spelling correction. These models consist of two components: a source model and a channel model. For many applications, people have devoted considerable energy to improving both components, with resulting improvements in overall system accuracy. However, relatively little research has gone into improving the channel model for spelling correction. This paper describes an improvement to noisy channel spelling correction via a more powerful model of spelling errors, be they typing mistakes or cognitive errors, than has previously been employed. Our model works by learning generic string to string edits, along with the probabilities of each of these edits. This more powerful model gives significant improvements in accuracy over previous approaches to noisy channel spelling correction.

1 Noisy Channel Spelling Correction

This paper will address the problem of automatically training a system to correct generic single word spelling errors.[1] We do not address the problem of correcting specific word set confusions such as {to,too,two} (see (Golding and Roth 1999)). We will define the spelling correction problem abstractly as follows: Given an alphabet Σ, a dictionary D consisting of strings in Σ^* and a string s, where $s \notin D$ and $s \in \Sigma^*$, find the word $w \in D$ that is most likely to have been erroneously input as s. The requirement that $s \notin D$ can be dropped, but it only makes sense to do so in the context of a sufficiently powerful language model.

all poss combos.

In a probabilistic system, we want to find $\text{argmax}_w P(w \mid s)$. Applying Bayes' Rule and dropping the constant denominator, we get the unnormalized posterior: $\text{argmax}_w P(s \mid w)^*P(w)$. We now have a noisy channel model for spelling correction, with two components, the source model P(w) and the channel model P(s | w). The model assumes that natural language text is generated as follows: First a person chooses a word to output, according to the probability distribution P(w). Then the person attempts to output the word w, but the noisy channel induces the person to output string s instead, according to the

[1] Two very nice overviews of spelling correction can be found in (Kukich 1992) and (Jurafsky and Martin 2000).

distribution P(s | w). For instance, under typical circumstances we would expect P(the | the) to be very high, P(teh | the) to be relatively high and P(hippopotamus | the) to be extremely low. In this paper, we will refer to the channel model as the *error model.*

Two seminal papers first posed a noisy channel model solution to the spelling correction problem. In (Mayes, Damerau et al. 1991), word bigrams are used for the source model. For the error model, they first define the *confusion set* of a string s to include s, along with all words w in the dictionary D such that s can be derived from w by a single application of one of the four edit operations:

(1) Add a single letter.
(2) Delete a single letter.
(3) Replace one letter with another.
(4) Transpose two adjacent letters.

Let C be the number of words in the confusion set of d. Then they define the error model, for all s in the confusion set of d, as:

$$P(s \mid d) = \begin{cases} \alpha \text{ if } s = d \\ \dfrac{(1-\alpha)}{(C-1)} \text{ otherwise} \end{cases}$$

This is a very simple error model, where α is the prior on a typed word being correct, and the remaining probability mass is distributed evenly among all other words in the confusion set.

Church and Gale (1991) propose a more sophisticated error model. Like Mayes, Damerau, et al. (1991), they consider as candidate source words only those words that are a single basic edit away from s, using the same edit set as above. However, two improvements are made. First, instead of weighing all edits equally, each unique edit has a probability associated with it. Second, insertion and deletion probabilities are conditioned on context. The probability of inserting or deleting a character is conditioned on the letter appearing immediately to the left of that character.

The error probabilities are derived by first assuming all edits are equiprobable. They use as a training corpus a set of space-delimited strings that were found in a large collection of text, and that (a) do not appear in their dictionary and (b) are no more than one edit away from a word that does appear in the dictionary. They iteratively run the spell checker over the training corpus to find corrections, then use these corrections to update the edit probabilities. Ristad and Yianilos (1997) present another algorithm for deriving these edit probabilities from a training corpus, and show that for the problem of word pronunciation, using the learned string edit distance gives one fourth the error rate compared to using unweighted edits.

2 An Improved Error Model

Previous error models have all been based on Damerau-Levenshtein distance measures (Damerau 1964; Levenshtein 1966), where the distance between two strings is the minimum number of single character insertions, substitutions and deletions (and in some cases, character pair transpositions) necessary to derive one string from another. Improvements have been made by associating probabilities with individual edit operations.

We propose a much more generic error model. Let Σ be an alphabet. Our model allows all edit operations of the form $\alpha \rightarrow \beta$, where $\alpha, \beta \in \Sigma^*$. $P(\alpha \rightarrow \beta)$ is the probability that when users intends to type the string α they type β instead. Note that the edit operations allowed in Church and Gale (1991), Mayes, Damerau et al. (1991) and Ristad and Yianilos (1997), are properly subsumed by our generic string to string substitutions.

In addition, we condition on the position in the string that the edit occurs in, $P(\alpha \rightarrow \beta \mid PSN)$, where PSN = {start of word, middle of word, end of word}.[2] The position is determined by the location of substring α in the source (dictionary) word. Positional information is a powerful conditioning feature for rich edit operations. For instance, P(e | a) does not vary greatly between the three positions mentioned above. However, P(ent | ant) is highly dependent upon position. People rarely mistype *antler* as *entler*, but often mistype *reluctant* as *reluctent*.

Within the noisy channel framework, we can informally think of our error model as follows. First, a person picks a word to generate. Then she picks a partition of the characters of that word. Then she types each partition, possibly erroneously. For example, a person might choose to generate the word *physical*. She would then pick a partition from the set of all possible partitions, say: **ph y s i c al**. Then she would generate each partition, possibly with errors. After choosing this particular word and partition, the probability of generating the string ***fisikle*** with the partition **f i s i k le** would be P(f | ph) *P(i | y) * P(s | s) *P(i | i) * P(k | c) *P(le | al).[3]

The above example points to advantages of our model compared to previous models based on weighted Damerau-Levenshtein distance. Note that neither P(f | ph) nor P(le | al) are modeled directly in the previous approaches to error modeling. A number of studies have pointed out that a high percentage of misspelled words are wrong due to a single letter insertion, substitution, or deletion, or from a letter pair transposition (Damerau 1964; Peterson 1986). However, even if this is the case, it does not imply that nothing is to be gained by modeling more powerful edit operations. If somebody types the string *confidant*, we do not really want to model this error as P(a | e), but rather P(ant | ent). And *anticedent* can more accurately be modeled by P(anti | ante), rather than P(i | e). By taking a more generic approach to error modeling, we can more accurately model the errors people make.

A formal presentation of our model follows. Let Part(w) be the set of all possible ways of partitioning string w into adjacent (possibly null) substrings. For a particular partition $R \in Part(w)$, where $|R|=j$ (R consists of j contiguous segments), let R_i be the i^{th} segment. Under our model,

$$P(s \mid w) = \sum_{R \in Part(w)} P(R \mid w) \sum_{\substack{T \in Part(s) \\ |T|=|R|}} \prod_{i=1}^{|R|} P(T_i \mid R_i)$$

One particular pair of alignments for s and w induces a set of edits that derive s from w. By only considering the best partitioning of s and w, we can simplify this to:

$$P(s \mid w) = \max_{R \in Part(w), T \in Part(s)} P(R|w) \prod_{i=1}^{|R|} P(T_i|R_i)$$

We do not yet have a good way to derive P(R | w), and in running experiments we determined that poorly modeling this distribution gave slightly worse performance than not modeling it at all, so in practice we drop this term.

3 Training the Model

To train the model, we need a training set consisting of $\{s_i, w_i\}$ string pairs, representing spelling errors s_i paired with the correct spelling of the word w_i. We begin by aligning the letters in s_i with those in w_i based on minimizing the edit distance

[2] Another good PSN feature would be *morpheme boundary*.

[3] We will leave off the positional conditioning information for simplicity.

able to insert empty ~~space~~ char ? inserted

between s_i and w_i, based on single character insertions, deletions and substitutions. For instance, given the training pair <akgsual, actual>, this could be aligned as:

```
    a   c     t   u   a   l
   /   /      \   \   \   \
  a   k   g   s   u   a   l
```

This corresponds to the sequence of edit operations:

a→a c→k ε→g t→s u→u a→a l→l

To allow for richer contextual information, we expand each nonmatch substitution to incorporate up to N additional adjacent edits. For example, for the first nonmatch edit in the example above, with N=2, we would generate the following substitutions:

c → k
ac → ak
c → kg
ac → akg
ct → kgs

We would do similarly for the other nonmatch edits, and give each of these substitutions a fractional count.

We can then calculate the probability of each substitution $\alpha \rightarrow \beta$ as count($\alpha \rightarrow \beta$)/count(α). count($\alpha \rightarrow \beta$) is simply the sum of the counts derived from our training data as explained above. Estimating count(α) is a bit tricky. If we took a text corpus, then extracted all the spelling errors found in the corpus and then used those errors for training, count(α) would simply be the number of times substring α occurs in the text corpus. But if we are training from a set of {s_i, w_i} tuples and not given an associated corpus, we can do the following:

(a) From a large collection of representative text, count the number of occurrences of α.

(b) Adjust the count based on an estimate of the rate with which people make typing errors.

Since the rate of errors varies widely and is difficult to measure, we can only crudely approximate it. Fortunately, we have found empirically that the results are not very sensitive to the value chosen.

Essentially, we are doing one iteration of the Expectation-Maximization algorithm (Dempster, Laird et al. 1977). The idea is that contexts that are useful will accumulate fractional counts across multiple instances, whereas contexts that are noise will not accumulate significant counts.

4 Applying the Model

Given a string s, where $s \notin D$, we want to return $\mathrm{argmax}_w\, P(w \mid s) P(w \mid context)$. Our approach will be to return an n-best list of candidates according to the error model, and then rescore these candidates by taking into account the source probabilities.

We are given a dictionary D and a set of parameters P, where each parameter is P($\alpha \rightarrow \beta$) for some $\alpha, \beta \in \Sigma^*$, meaning the probability that if a string α is intended, the noisy channel will produce β instead. First note that for a particular pair of strings {s, w} we can use the standard dynamic programming algorithm for finding edit distance by filling a $|s|*|w|$ weight matrix (Wagner and Fisher 1974; Hall and Dowling 1980), with only minor changes. For computing the Damerau-Levenshtein distance between two strings, this can be done in O($|s|*|w|$) time. When we allow generic edit operations, the complexity increases to O($|s|^2*|w|^2$). In filling in a cell (i,j) in the matrix for computing Damerau-Levenshtein distance we need only examine cells (i,j-1), (i-1,j) and (i-1,j-1). With generic edits, we have to examine all cells (a,b) where $a \le i$ and $b \le j$.

We first precompile the dictionary into a trie, with each node in the trie

corresponding to a vector of weights. If we think of the x-axis of the standard weight matrix for computing edit distance as corresponding to w (a word in the dictionary), then the vector at each node in the trie corresponds to a column in the weight matrix associated with computing the distance between s and the string prefix ending at that trie node.

We store the $\alpha \rightarrow \beta$ parameters as a trie of tries. We have one trie corresponding to all strings α that appear on the left hand side of some substitution in our parameter set. At every node in this trie, corresponding to a string α, we point to a trie consisting of all strings β that appear on the right hand side of a substitution in our parameter set with α on the left hand side. We store the substitution probabilities at the terminal nodes of the β tries.

By storing both α and β strings in reverse order, we can efficiently compute edit distance over the entire dictionary. We process the dictionary trie from the root downwards, filling in the weight vector at each node. To find the substitution parameters that are applicable, given a particular node in the trie and a particular position in the input string s (this corresponds to filling in one cell in one vector of a dictionary trie node) we trace up from the node to the root, while tracing down the α trie from the root. As we trace down the α trie, if we encounter a terminal node, we follow the pointer to the corresponding β trie, and then trace backwards from the position in s while tracing down the β trie.

Note that searching through a static dictionary D is not a requirement of our error model. It is possible that with a different search technique, we could apply our model to languages such as Turkish for which a static dictionary is inappropriate (Oflazer 1994).

Given a 200,000-word dictionary, and using our best error model, we are able to spell check strings not in the dictionary in approximately 50 milliseconds on average, running on a Dell 610 500mhz Pentium III workstation.

5 Results

5.1 Error Model in Isolation

We ran experiments using a 10,000-word corpus of common English spelling errors, paired with their correct spelling. We used 80% of this corpus for training and 20% for evaluation. Our dictionary contained approximately 200,000 entries, including all words in the test set. The results in this section are obtained with a language model that assigns uniform probability to all words in the dictionary. In Table 1 we show K-best results for different maximum context window sizes, without using positional information. For instance, the 2-best accuracy is the percentage of time the correct answer is one of the top two answers returned by the system. Note that a maximum window of zero corresponds to the set of single character insertion, deletion and substitution edits, weighted with their probabilities. We see that, up to a point, additional context provides us with more accurate spelling correction and beyond that, additional context neither helps nor hurts.

Max Window	1-Best	2-Best	3-Best
0	87.0	93.9	95.9
CG	89.5	94.9	96.5
1	90.9	95.6	96.8
2	92.9	97.1	98.1
3	93.6	97.4	98.5
4	93.6	97.4	98.5

Table 1 Results without positional information

In Table 1, the row labelled CG shows the results when we allow the equivalent set of edit operations to those used in (Church and Gale 1991). This is a

proper superset of the set of edits where the maximum window is zero and a proper subset of the edits where the maximum window is one. The CG model is essentially equivalent to the Church and Gale error model, except (a) the models above can posit an arbitrary number of edits and (b) we did not do parameter reestimation (see below).

Next, we measured how much we gain by conditioning on the position of the edit relative to the source word. These results are shown in Table 2. As we expected, positional information helps more when using a richer edit set than when using only single character edits. For a maximum window size of 0, using positional information gives a 13% relative improvement in 1-best accuracy, whereas for a maximum window size of 4, the gain is 22%. Our full strength model gives a 52% relative error reduction on 1-best accuracy compared to the CG model (95.0% compared to 89.5%).

Max Window	1-Best	2-Best	3-Best
0	88.7	95.1	96.6
1	92.8	96.5	97.4
2	94.6	98.0	98.7
3	95.0	98.0	98.8
4	95.0	98.0	98.8
5	95.1	98.0	98.8

Table 2 Results with positional information.

We experimented with iteratively reestimating parameters, as was done in the original formulation in (Church and Gale 1991). Doing so resulted in a slight degradation in performance. The data we are using is much cleaner than that used in (Church and Gale 1991) which probably explains why reestimation benefited them in their experiments and did not give any benefit to the error models in our experiments.

5.2 Adding a Language Model

Next, we explore what happens to our results as we add a language model. In order to get errors in context, we took the Brown Corpus and found all occurrences of all words in our test set. Then we mapped these words to the incorrect spellings they were paired with in the test set, and ran our spell checker to correct the misspellings. We used two language models. The first assumed all words are equally likely, i.e. the null language model used above. The second used a trigram language model derived from a large collection of on-line text (not including the Brown Corpus). Because a spell checker is typically applied right after a word is typed, the language model only used left context.

We show the results in Figure 1, where we used the error model with positional information and with a maximum context window of four, and used the language model to rescore the 5 best word candidates returned by the error model. Note that for the case of no language model, the results are lower than the results quoted above (e.g. a 1-best score above of 95.0%, compared to 93.9% in the graph). This is because the results on the Brown Corpus are computed per *token*, whereas above we were computing results per *type*.

One question we wanted to ask is whether using a good language model would obviate the need for a good error model. In Figure 2, we applied the trigram model to resort the 5-best results of the CG model. We see that while a language model improves results, using the better error model (Figure 1) still gives significantly better results. Using a language model with our best error model gives a 73.6% error reduction compared to using a language model with the CG error model. Rescoring the 20-best output of the CG model instead of the 5-best only improves the 1-best accuracy from 90.9% to 91.0%.

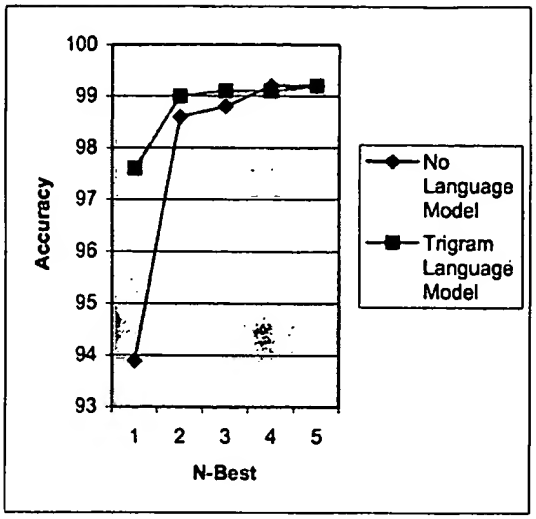

Figure 1 Spelling Correction Improvement When Using a Language Model

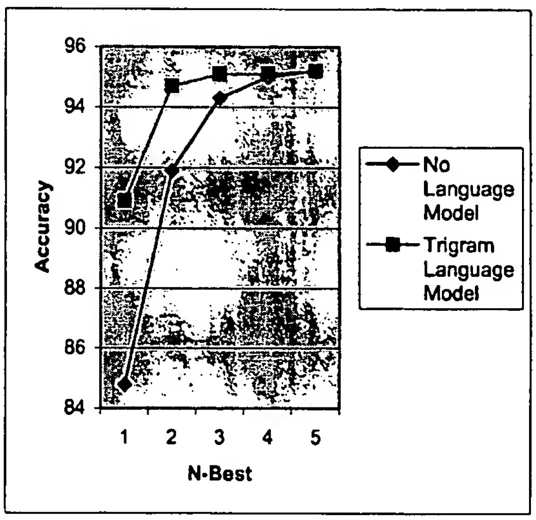

Figure 2 Using the CG Error Model with a Trigram Language Model

Conclusion

We have presented a new error model for noisy channel spelling correction based on generic string to string edits, and have demonstrated that it results in a significant improvement in performance compared to previous approaches. Without a language model, our error model gives a 52% reduction in spelling correction error rate compared to the weighted Damerau-Levenshtein distance technique of Church and Gale. With a language model, our model gives a 74% reduction in error.

One exciting future line of research is to explore error models that adapt to an individual or subpopulation. With a rich set of edits, we hope highly accurate individualized spell checking can soon become a reality.

References

Church, K. and W. Gale (1991). "Probability Scoring for Spelling Correction." Statistics and Computing 1: 93-103.

Damerau, F. (1964). "A technique for computer detection and correction of spelling errors." Communications of the ACM 7(3): 659-664.

Dempster, A., N. Laird, et al. (1977). "Maximum likelihood from incomplete data via the EM algorithm." Journal of the Royal Statistical Society 39(1): 1-21.

Golding, A. and D. Roth (1999). "A Winnow-Based Approach to Spelling Correction." Machine Learning 34: 107-130.

Hall, P. and G. Dowling (1980). "Approximate string matching." ACM Computing Surveys 12(4): 17-38.

Jurafsky, D. and J. Martin (2000). Speech and Language Processing, Prentice Hall.

Kukich, K. (1992). "Techniques for Automatically Correcting Words in Text." ACM Computing Surveys 24(4): 377-439.

Levenshtein, V. (1966). "Binary codes capable of correcting deletions, insertions and reversals." Soviet Physice -- Doklady 10: 707-710.

Mayes, E., F. Damerau, et al. (1991). "Context Based Spelling Correction." Information Processing and Management 27(5): 517-522.

Oflazer, K. (1994). Spelling Correction in Agglutinative Languages. Applied Natural Language Processing, Stuttgart, Germany.

Peterson, J. (1986). "A note on undetected typing errors." Communications of the ACM **29**(7): 633-637.

Ristad, E. and P. Yianilos (1997). Learning String Edit Distance. International Conference on Machine Learning, Morgan Kaufmann.

Shannon, C. (1948). "A mathematical theory of communication." Bell System Technical Journal **27**(3): 379-423.

Wagner, R. and M. Fisher (1974). "The string to string correction problem." JACM **21**: 168-173.

An SVG Case Study: Integrated, Dynamic Avalanche Forecasting
by Chris Cochella, Tyler Cruickshank | Pages: 1, **2**

SVG provides a tag to uniquely identify a group of visual elements. Here we use SVG's group ("g") tag to give the elements that define our graphic a name ("id"). The name that defines our graphic will be used by our JavaScript event handler. The next thing we want to point out is SVG's handy `transform` attribu provides a way to scale or resize your object or entire group of graphic objects. attribute lets us move the graphic to a particular location within the coordinate s we doing this? In AMT we have a rectangular view port within which we displa graphs of the weather stations. The individual weather station SVG graphics ha and coordinate system; we must `transform` the size and positioning of the grap is added to the AMT, it fits inside the rectangular view port. This means that yo about any SVG graphic anywhere you want.

```
<g id="altaGraph" width="6000" height="4500"
  transform="scale(.104,.0973) translate(2625,1230)">
```

Then we draw the various graphic primitives such as the outer chart box:

```
<rect x="0" y="0" width="6000" height="4500"
  fill="#dcdcdc" stroke="black" stroke-width="20"/>
```

We place a trash can image for disposing or removing a particular chart with th `onclick="snowTkMetRemove()"` function (this is the really exciting part which step 5 below):

```
<image xlink:href="./images/trash.gif" x="5700" y="50"
width="300" height="300" style="display:inline"onclick="snowTkMe
```

Once the base layer of graphical components are constructed, and we have prep a simple matter of outputing the data like the Temperature line segments (blue l

```
<polyline fill="none" stroke="red" stroke-width="7"
  points="400,1533.33 431.13,1533.33"/>
```

There are many other lines, rectangles, and text values which are placed on the displayed here. For more information on SVG details refer to SVG Essentials b Eisenberg.

We have automated the chart generation process to follow the data harvesting. T additional step for chart creation: compression. The Adobe SVG plug in permit of the source files. This is useful in our case because the SVG text files are appr

What are Web Services?
What are XForms?
XSLT Recipe of the Day

Manage Your Account
Forgot Your Password?

▶ FIND
Search
Article Archive

▶ COLUMNS
<taglines/>
Dive into XML
Hacking Congress
Hacking the Library
Java Web Services
Security
Jon Udell
Perl and XML
Practical XQuery
Python and XML
Rich Salz
Sacré SVG
Standards Lowdown
The Restful Web
Transforming XML
XML Q&A
XML Tourist
XML-Deviant

▶ GUIDES
XML Resources
Buyer's Guide
Events Calendar
Standards List
Submissions List

▶ TOOLBOX
Syntax Checker

Kilobytes. That's not huge, but when you are trying to get out the door, time is o zipping these files with `gzip` we can gain an order of magnitude reduction in fil example, one chart is 199955 Kilobytes before compression and 18247 Kilobyt relative lightweightedness is one of the hidden benefits of SVG over other vecto

Now that the data has been harvested and the charts have been generated, we ne the charts into a single viewer.

Step 5: Where it all comes together

Each weather station is a separate SVG file and presentation. However, for our to integrate each station into a regional Toolkit so that the user can jump from o another and see a consistent output of relevant information. In a sense, the Tool information appliance used for displaying external weather data.

For the toolkit interface, you can see that we employ some of the same SVG dis creating a box, adding an image and a transparent overlay to create a list of "Me Graphs" shown in Figure 3 and in the code sample below:

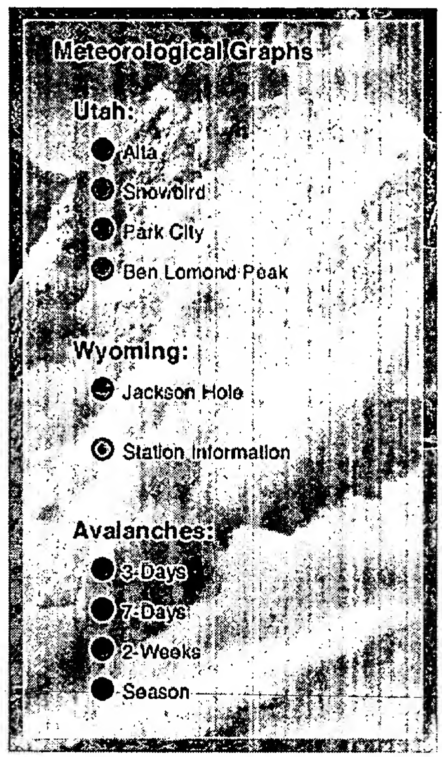

Figure 3: List of Graphs


```
<g id="smallWindowTemplate">
<desc> The background Template </desc>

<image xlink:href="./images/avalanchel.jpg" x="35"
  y="120" width="220" height="280"style="display:inline"/>

<rect x="45" y="130" width="200"height="260"
  fill="yellow" opacity=".6"stroke="none" stroke-width="5"/>
```

```
<rect x="35" y="120" width="220"height="280"
  fill="none" stroke-opacity=".4"stroke="red"
  stroke-width="3"/>

<text style='font-size:12pt; font-weight:bold;
  font-family:sans-serif;fill:black; text-anchor:center;'
  x="60"y="150">Meteorological Graphs</text>

<text style='font-size:12pt; font-weight:bold;
  font-family:sans-serif;fill:black; text-anchor:center;'
  x="70"y="180">Utah:</text>

</g>
```

We then add each weather graph choice as a radio button:

```
<g id="messageWindow">

<text style='font-size:10pt; font-family:sans-serif;
  fill:black;text-anchor:left;'
  x="95" y="200">Alta</text>

<circle id="wxla" cx="85" cy="195" r="5"
  fill="blue" stroke="black" stroke-width="2"
  onclick="snowTkMetALTA(evt)" />

</g>
```

Aside from the basic visual elements, we have included an ECMAScript `oncli`
where we take advantage of Adobe's SVG plugin server-connection capabilities
data sources. In this case, our external data is the station-specific SVG graphs w
steps 3 through 4. This functionality is very nice because once we have develop
appliance we no longer have to touch it, even though the data is constantly chan
refer to that external data source, which also means we download only what is r
shows a specific weather station graph within the complete Toolkit interface.

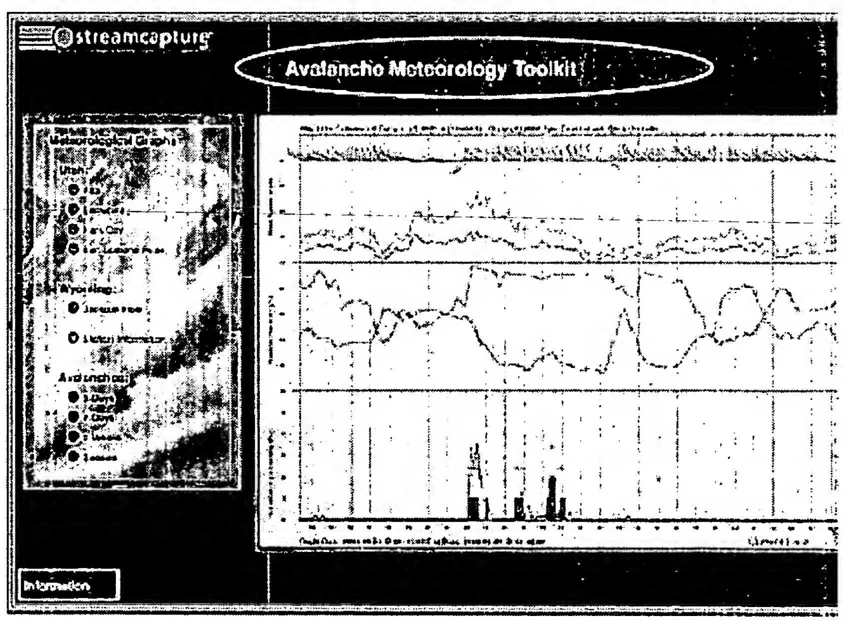

Figure 4: The Completed Toolkit

This process of loading and unloading external data sources requires three meth retrieve the external SVG file with the `getURL` method, another to act as a callb parses, returns, and inserts the XML into the main SVG document tree, and a fi destroy or delete the inserted XML when the trash icon is selected (see the `snow` method in the above code listing that adds the trash can image). These 3 method along with the beginning of the SVG document for the Toolkit interface.

```
<?xmlversion="1.0"?>

<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 20010904//EN"
  "http://www.w3.org/TR/2001/REC-SVG-20010904/DTD/svg10.dtd">

<svg width="900" height="900" viewBox="0 0900 900"
  preserveAspectRatio="xMinYMin" >

<!-- ECMAScript -->

<script type='text/ecmascript'><![CDATA[

// Method calling the getURL()

function snowTkMetALTA (evt)
{
  var node = evt.target.ownerDocument.getElementById("altaGraph"
  node=evt.getTarget;
  var url='./altaGraph.svgz';
  getURL(url,snowTkMetCallbackALTA);
}

function snowTkMetCallbackALTA (document)
{
  fragment=parseXML(document.content,document);
  document.getElementById("messageWindow").appendChild(fragment)
}

function snowTkMetRemoveAlta (evt)
{
  var node = evt.target.ownerDocument.getElementById("altaGraph"
  node.parentNode.removeChild(node);
}

</script>
```

There are a couple of items to take note of in the ECMAScript. First, notice that directory path to our weather station graphic. The method `getURL()` retrieves it it on to our AMT in the `snowTkMetCallbackALTA` function. Second, the `getEle` method is given two different arguments: "messageWindow" and "altaGraph". T "messageWindow" argument is the group id of the location in our AMT where our weather graph. Now, the group "messageWindow" will have a new element group of elements that defines our weather graph. "altaGraph" is the group id of station SVG graphic that we want to add.

This is where giving names to graphic elements is important. The "altaGraph" g scale and translate information that allows the graphic to fit inside the view port

appended to "messageWindow". Not only is it important for placing or inserting graph, it is also important for specifying what portion of the document to remov information, see Antoine Quint's article SVG Tips and Tricks: Adobe's SVG Vi XML.com.) When we are done with each chart, we simply remove it from the d add a new one-very clean.

Summary

The primary benefits of our approach are the integration of disparate informatio single, lightweight display of relevant real-time information. Perl, or course, is g processing. SVG along with Adobe's server-connection capability has proven to and lightweight means of information display. We are planning the integration o dynamic information like avalanche pictures posted by the public, integration o Avalanche Center forecast advisory, and a regional map of recent avalanches. B this information in one place, backcountry travelers and avalanche forecasters h quickly assess backcountry conditions and make appropriate, informed decision to the possibilities for creating dynamic, lightweight displays of real-time data: monitoring devices, traffic monitoring, manufacturing processes, financial data,

Resources

- Avalanche.org
- W3C SVG
- Adobe's SVG Site
- SVG at oreillynet.com

Leave your comments or questions on this article in our forum.
(You must be a member of XML.com to use this feature.)*

Comment on this Article

Titles Only Titles Only Newest First

- Non-proprietory SVG is even better!
  2003-04-24 12:09:38 Robert McKinnon [Reply]

  Because the Avalanche Meteorology Toolkit (AMT) does not strictly adh specification[1] I am unable to use it with the Squiggle SVG Browser. Str Web standards[2] reduces the cost and complexity of development while accessibility and long-term viability of any site published on the Web.

  I am a Linux user who views SVG content using the Squiggle SVG Brow with Apache's Batik SVG toolkit.[3]

  I have found two reasons why AMT does not work on Squiggle:

1) AMT's SVG does not match the SVG 1.1 specification in a few instanc 'left' and 'center' are not valid values for the text-anchor property. The val specification are 'start' and 'middle'.[4]

2) AMT makes use of proprietory Adobe SVG Viewer features that are n 1.1 specification. Rather than using Adobe's server-connection capabilitie data sources, xlink:href links in the menu could be applied to achieve the click on the menu would then request new pages from the server, thus ma traditional HTTP GET request to retrieve new data views.

As it stands the AMT looks like a great toolkit, however currently it's a p SVG Viewer application. With a little work it could be made into an awe standard, cross-platform compatible Web application, by strictly adhering Web standards.

[1] http://www.w3.org/TR/SVG11/
[2] http://www.webstandards.org/about/
[3] http://xml.apache.org/batik/
[4] http://www.w3.org/TR/SVG11/text.html#TextAnchorProperty

- Non-proprietory SVG is even better!
  2003-04-25 14:53:37 Tyler Cruickshank [Reply]

  Robert,

  Thanks very much for giving our article a read and for providing so thought.

  In short, we agree with your comments. SVG is open-source so we away from any proprietary features.
  On the other hand, when we originally built the toolkit a year and a simply wanted to get it done. Most of our users prefer the easiest m the SVG, therefore, the Adobe viewer was the easiest solution. Hav that we have the toolkit functioning and are more familiar with dyn SVG we should and intend to go back and remove the proprietary f

  Down the road, if we keep our SVG adhering to the W3C standards worry what viewer is being used.

  Thanks for reminding us.

  -tyler